A Randomized Rounding Algorithm for the Asymmetric Traveling Salesman Problem

Michel X. Goemans   Nicholas J. A. Harvey   Kamal Jain   Mohit Singh

Abstract

We present an algorithm for the asymmetric traveling salesman problem on instances which satisfy the triangle inequality. Like several existing algorithms, it achieves approximation ratio $O(\log n)$. Unlike previous algorithms, it uses randomized rounding.

1 Introduction

Let V be a set of n vertices and let $c : V \times V \to \mathbb{R}_+$ be a cost function. We assume the triangle inequality: $c_{i,j} \leq c_{i,k} + c_{k,j}$ for all vertices i, j, k. The asymmetric traveling salesman problem (ATSP) is to solve $\min_\pi \sum_{v \in V} c_{v,\pi(v)}$ over all cyclic permutations π on V. A subgraph $\{ (v, \pi(v)) : v \in V \}$ is called a tour; we seek a minimum cost tour.

We will use the following standard notation. For $U \subseteq V$, define

$$\begin{aligned} \delta^+(U) &= \{ (v, w) : v \in U,\ w \notin U \}, \\ \delta^-(U) &= \{ (w, v) : v \in U,\ w \notin U \}. \end{aligned}$$

For any vector $x \in \mathbb{R}_+^{V \times V}$ and $F \subseteq V \times V$, we use the notation $x(F) = \sum_{e \in F} x_e$.

The Held-Karp linear programming relaxation of ATSP is as follows.

$$\begin{array}{llll} \min & c^{\mathsf{T}} x & & \\ \text{s.t.} & x(\delta^-(\{v\})) & = & x(\delta^+(\{v\})) \quad \forall v \in V \\ & x(\delta^+(U)) & \geq & 1 \qquad\qquad\quad\ \forall \emptyset \neq U \subsetneq V \\ & x & \geq & 0 \end{array}$$

By standard shortcutting arguments, we may assume that $x_{(v,w)} \leq 1$ for all v, w, and that $x(\delta^+(\{v\})) = 1$ for all v.

Several polynomial-time algorithms [3] [9, pp. 125] [4] [2] are known for computing a tour whose cost is at most a factor $O(\log n)$ larger than the optimum. In addition, several proofs [10, 8] are known showing that the integrality gap of the Held-Karp relaxation is $O(\log n)$. This note provides another such algorithm and another such proof of the integrality gap.

2 The Algorithm

The algorithm proceeds in two steps. In the first step, we round the fractional solution using a simple randomized rounding schema to obtain *nearly-balanced* graph. In the second step, we solve the *patch up* problem to make the graph Eulerian. The algorithm succeeds in returning a connected Eulerian subgraph of small cost with high probability.

2.1 Constructing a nearly balanced graph

Let x be any feasible solution to this linear program. Since x is balanced at every vertex, this implies that x is Eulerian, i.e., $x(\delta^+(U)) = x(\delta^-(U))$ for all $U \subseteq V$. So x is a fractional solution for which all cuts are perfectly balanced. We now use x to construct an integral solution z for which all cuts are nearly-balanced, i.e., for each cut $U \subsetneq V$, $\frac{z(\delta^+(U))}{z(\delta^-(U))} \leq 2$. Moreover, the cost of z is at most $O(\log n) \cdot c^\mathsf{T} x$.

The first observation is that x has equivalent cut values to an undirected graph. Formally, for $U \subseteq V$, define $\delta(U) = \{\ \{v, w\}\ :\ v \in U,\ w \notin U\ \}$. For $y \in \mathbb{R}_+^{\binom{V}{2}}$ and $F \subseteq \binom{V}{2}$, let $y(F) = \sum_{e \in F} y_e$.

Claim 1. Since x is Eulerian, there exists $y \in \mathbb{R}_+^{\binom{V}{2}}$ such that $y(\delta(U)) = x(\delta^+(U))$ for all $U \subseteq V$.

Proof. Define $y_{\{v,w\}} = (x_{v,w} + x_{w,v})/2$ for all v, w. Then

$$y(\delta(U)) \;=\; \sum_{v \in U,\, w \notin U} \frac{x_{v,w} + x_{w,v}}{2} \;=\; \frac{1}{2}\Big(x(\delta^+(U)) + x(\delta^-(U))\Big) \;=\; x(\delta^+(U)),$$

as required. $\square$

We now apply a random sampling result of Karger [5]. For convenience, we reprove it here in our notation. For any undirected graph with minimum cut value c, Karger [5] shows that the number of cuts of value at most αc is less than $\binom{n}{2\alpha}$. This result applies to the graph induced by y and hence, by Claim 1, also to x:

$$\left|\left\{\ U\ :\ \emptyset \neq U \subsetneq V,\ x(\delta^+(U)) \leq \alpha\ \right\}\right| \;\leq\; n^{2\alpha}. \tag{1}$$

To round x, we must first scale it so that its minimum cut value is large. Let G be the directed, weighted, multigraph obtained from x by taking $K := 100 \ln n$ parallel copies of each edge, each of the same weight as in x. Let c_i be the value of the i^{th} cut, ordered such that $K \leq c_1 \leq c_2 \leq \cdots$. We will construct a directed multigraph H by taking each edge of G with probability proportional to its weight. The expected number of edges chosen by H in the i^{th} cut is c_i. Let p_i be the probability that the actual number of edges chosen in the i^{th} cut diverges from its expectation by more than an ϵ fraction. By a Chernoff bound, $p_i \leq 2e^{-\epsilon^2 c_i/3}$.

We will ensure that no cut diverges significantly from its expectation by choosing ϵ appropriately and applying a union bound. Define $\epsilon = \sqrt{1/10}$. Since $c_i \geq K = 100 \ln n$, we have $p_i \leq n^{-3}$ for all i. For the small cuts, we use the bound

$$\sum_{i=1}^{2n^2} p_i \;=\; O(1/n). \tag{2}$$

For the large cuts, we use a different bound. Eq. (1) implies that $c_{n^{2\alpha}} \geq \alpha K$. Letting $i = n^{2\alpha}$, we have $c_i \geq K \ln i/(2 \ln n)$, and hence $p_i \leq i^{-3/2}$. Thus

$$\sum_{i > n^2} p_i \;\leq\; \sum_{i > n^2} i^{-3/2} \;<\; \int_{n^2}^{\infty} x^{-3/2}\, dx \;=\; O(1/n). \tag{3}$$

Combining Eq. (2) and Eq. (3) shows that with probability $1 - O(1/n)$, no cut in H diverges from its expectation by more than an ϵ fraction. The expected cost of H is $K \cdot c^{\mathsf{T}}x$, so a Chernoff bound again implies that the cost of H is $O(\log n) \cdot c^{\mathsf{T}}x$ with high probability.

Let $z \in \mathbb{Z}_+^{V \times V}$ be the vector giving the total weight of the edges in the multigraph H. Assuming that no cut in H diverges significantly from its expectation, we have

$$\frac{z(\delta^+(U))}{z(\delta^-(U))} \;\leq\; \frac{1+\epsilon}{1-\epsilon} \;\leq\; 2 \qquad \forall\, \emptyset \neq U \subsetneq V. \tag{4}$$

The last inequality follows because $\epsilon < 1/3$. Thus z is a nearly-balanced graph with high probability.

2.2 Patching Up

We now make z Eulerian by "patching it up" with another graph w. That is, we seek another vector $w \in \mathbb{Z}_+^{V \times V}$ such that $z + w$ is connected and Eulerian — an integral, feasible solution to the Held-Karp relaxation of ATSP. Indeed, we show that a subgraph of z can be used to patch up z.

Consider the transshipment problem on V where each vertex v has demand $b(v) := z(\delta^+(v)) - z(\delta^-(v))$. Hoffman's circulation theorem [7, Corollary 11.2f] implies that there exists a subgraph of z giving a feasible, integral transshipment for these demands iff the capacity of each cut is at least its demand:

$$z(\delta^-(U)) \;\geq\; \sum_{v \in U} b(v) \;=\; z(\delta^+(U)) - z(\delta^-(U)).$$

This inequality is implied by Eq. (4), so the desired transshipment w exists, and its cost is at most $c^{\mathsf{T}}z$. Thus $z + w$ gives a connected, Eulerian graph of cost at most $2\,c^{\mathsf{T}}z$, which is $O(\log n) \cdot c^{\mathsf{T}}x$, as argued above. By shortcutting, we obtain a tour of no worse cost. If x is an optimum solution of the linear program then the resulting tour is at most a factor $O(\log n)$ larger than the optimum tour. Consequently, the integrality gap of this linear program is at most $O(\log n)$.

3 Tight Example

We now show that the analysis of the algorithm given above is tight to within constant factors — we give an example where we must choose K to be $\Omega(\log n)$ in the first step of the algorithm. This condition is necessary not only to ensure the nearly-balanced condition but also to ensure that H is connected.

Consider any extreme point x such that $x_a < \frac{2}{3}$ for every arc $a \in A$ and let E be the support of x. Such extreme points exist of arbitrarily large size [1]. Using the fact that $|E| < 3n - 2$ where $n = |V|$, we obtain that there in an independent set of vertices V_1 of size at least $\frac{n}{6}$. Since $x_a < \frac{2}{3}$ for each $a \in A$ and $x(\delta^-(v)) + x(\delta^+(v)) = 2$, the probability that v is a not an isolated vertex in H is at most $1 - \frac{1}{27^K}$ for each $v \in V_1$. Since V_1 is an independent set, these events are independent. Hence, the probability that none of the vertices in V_1 is an isolated vertex in H is at most $(1 - \frac{1}{27^K})^{\frac{n}{6}}$. In order for H to be connected with constant probability, we must take $K = \Omega(\log n)$.

References

[1] M. Charikar, M. Goemans and H. Karloff. "On the Integrality Ratio for the Asymmetric Traveling Salesman Problem", Mathematics of Operations Research, 31, pp. 245–252, 2006.

[2] Uriel Feige and Mohit Singh. "Improved Approximation Ratios for Traveling Salesperson Tours and Paths in Directed Graphs". Proceedings of the 10th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems (APPROX), 2007.

[3] A.M. Frieze, G. Galbiati, and F. Maffioli. "On the Worst-Case Performance of Some Algorithms for the Asymmetric Traveling Salesman Problem". Networks, 12(1):23-39, 1982.

[4] H. Kaplan, M. Lewenstein, N. Shafrir and M. Sviridenko. "Approximation algorithms for asymmetric TSP by decomposing directed regular multigraphs". Journal of the ACM, 52(4):602–626, 2005.

[5] David R. Karger. "Random Sampling in Cut, Flow, and Network Design Problems". Mathematics of Operations Research, 24(2):383-413, 1999.

[6] David R. Karger. "Minimum Cuts in Near-Linear Time". Journal of the ACM, 47(1):46–76, 2000.

[7] Alexander Schrijver. "Combinatorial Optimization: Polyhedra and Efficiency". Springer, 2004.

[8] David B. Shmoys and David P. Williamson. "Analyzing the Held-Karp TSP bound: a monotonicity property with application". Information Processing Letters, 35(6):281-285, 1990.

[9] David P. Williamson. "Lecture Notes on Approximation Algorithms". IBM Research Report RC 21409, 1999.

[10] L. Wolsey. "Heuristic Analysis, Linear Programming and Branch and Bound". Mathematical Programming Study, 13:121-134, 1980.